

# REPORT DOCUMENTATION PAGE

*Form Approved  
OMB No. 0704-0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington Headquarters Service, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.

**PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

1. REPORT DATE (DD-MM-YYYY)	2. REPORT DATE 9/1/99	3. DATES COVERED (From - To) 4/1/98 - 8/31/99		
4. TITLE AND SUBTITLE  A Theory of Hierarchical, Distributed Systems		5a. CONTRACT NUMBER		
		5b. GRANT NUMBER N00014-98-1-0585		
		5c. PROGRAM ELEMENT NUMBER		
6. AUTHOR(S)  Professor Pravin Varaiya		5d. PROJECT NUMBER		
		5e. TASK NUMBER		
		5f. WORK UNIT NUMBER		
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)  University of California, Berkeley Electronics Research Laboratory 253 Cory Hall, Berkeley, CA 94720		8. PERFORMING ORGANIZATION REPORT NUMBER  442427-23070		
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)  Office of Naval Research, Program Officer Allen Moshfegh ONR 351, Ballston Centre Tower One 800 North Quincy Street Arlington, VA 22217-5660		10. SPONSOR/MONITOR'S ACRONYM(S)		
		11. SPONSORING/MONITORING AGENCY REPORT NUMBER		
12. DISTRIBUTION AVAILABILITY STATEMENT  <b>19991004 324</b>				
13. SUPPLEMENTARY NOTES				
14. ABSTRACT  The research advanced in three domains: development of an architecture for autonomous, intelligent vehicles; software environment for detailed specification of the control structure and simulation; and detailed system design for several missions. The tangible products of the research at the end of the 15 months are: Architectures and detailed design and simulation of complex missions in SHIFT; A more general function architecture, conceptual building blocks and initial specification of corresponding SHIFT classes; A SHIFT "port" to the NT and Windows platform.				
15. SUBJECT TERMS  hierarchical control architectures				
16. SECURITY CLASSIFICATION OF:  a. REPORT unlimited		17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON Prof. Pravin Varaiya
				19b. TELEPHONE NUMBER (Include area code) 510-642-5270

## **Theory of hierarchical, distributed systems**

***PI. Pravin Varaiya***

***Progress Report for Period 5/98 – 8/99***

### **Summary**

This is the progress report of the project, *Theory of Hierarchical, Distributed Systems*, for the 15-month period May 1998 - August 1999.

Research advanced in three domains: development of an architecture for autonomous, intelligent vehicles; software environment for detailed specification of the control structure and simulation; and detailed system design for several missions.

The tangible products of the research at the end of the 15 months are:

1. Architectures and detailed design and simulation of complex missions in SHIFT;
2. A more general functional architecture, conceptual building blocks and initial specification of corresponding SHIFT classes;
3. A SHIFT "port" to the NT and Windows platforms.

The report concludes with a brief statement of plans for the next period:

1. An extension of SHIFT to model "learning" agents;
2. Design of more "deliberative" agents.

### **Objective**

This project has three goals:

1. The development of hierarchical control architectures for distributed, autonomous systems of the kind envisaged by ONR;
2. The development of software tools to assist the design, simulation, analysis, and implementation of those control architectures;
3. Working with other teams in this program to actually design, simulate, and demonstrate these systems at least at the level of "proof of concept."

We are meeting goals 1 and 2 above. So far as goal 3 is concerned, we are successfully collaborating with the team led by Professor Sastry. We are not yet collaborating with any other team.

### **Setting**

The current wave of research and development has led to many advances in the capability of *individual* systems. We now have smart vehicles, smart guidance, and smart sensors. These systems are "smart" because they can diagnose themselves and adapt to external changes. As a result, they can carry out an assigned task with minimal human supervision. A smart systems has the ability to be an autonomous agent.

The next wave of research must seek to advance intelligence to the level of *systems* of autonomous agents. Figure 1 depicts an operational scenario with many autonomous agents. Each agent has a limited and specialized capability. But the agents are organized in such a way that collectively they have the capability to carry out complex missions. *The intelligence of the collectivity lies in the organization.*

#### **Figure 6. Survey, tag, and rescue**

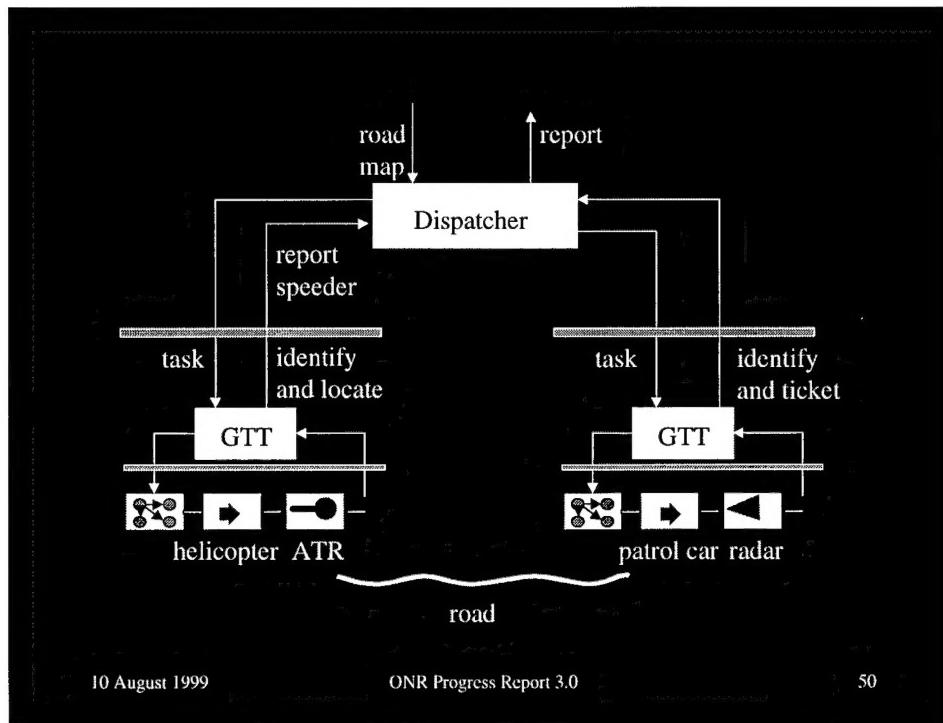
### **Simulation and design**

The success with the examples briefly described above reinforce our conviction that a small set of agents can be designed that can combined together to carry out complex missions. The missions described above are all simulated in SHIFT, which is an excellent platform for research in hierarchical architectures.

### **Future plans**

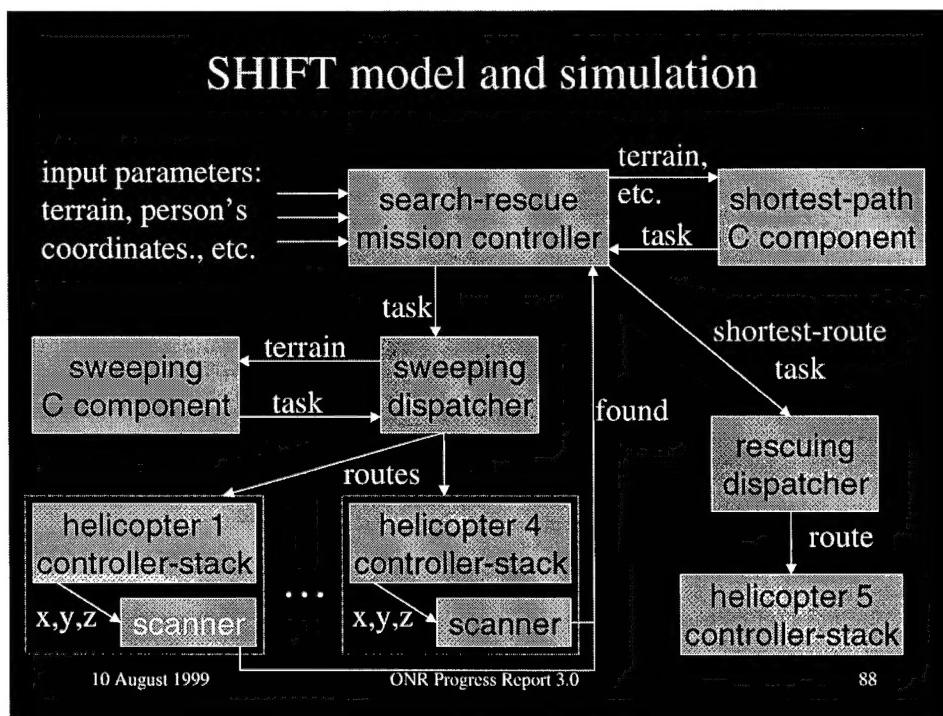
We will pursue two research directions over the next year. The first concerns a richer specification of agents so that they can be truly "deliberative" instead of simply the executors of tasks assigned by superiors. This will be done by requiring agents to provide an "estimate" of how well it can do a task. The superior can now weight different ways of carrying out a task. The second direction concerns the development of agents that can change its specification dynamically as it gains experience.

- **Survey and tag:** In this more complex scenario, a speeding vehicle is identified by a helicopter and reported. The dispatcher then assigns an available patrol car to tag the speeding vehicle.



**Figure 5: Survey and tag**

- **Survey, tag, and rescue:** In this mission autonomous helicopters survey a region, locate a distressed person, and then guide a rescue helicopter to rescue the person.



and networks that link agents will change. One way to conceptualize this is to imagine that the links in Figure 2 get redrawn as a mission unfolds.

## **Conceptual and SHIFT building blocks**

The scenario of Figure 1 can be created out of five conceptual building blocks:

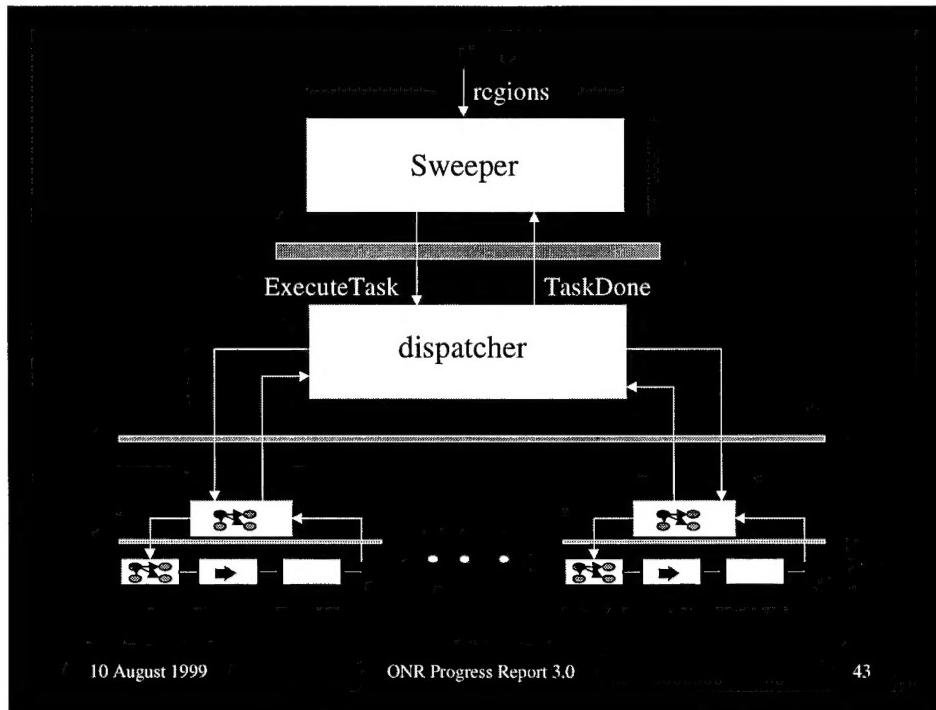
- Agents
  - receive tasks from and report to superior agents
  - task subordinate agents and control equipment
  - receive reports (from agents) and data (from sensors)
- Equipment—vehicles that move and sensors that “see”
  - Follow commands from low layer controllers (agents)
  - Generate movement and data
- Terrain—the space in which agents and equipment are located
- Networks—support communication among agents and equipment
- Groups—agents that cooperate and share information

In our experience, these five building blocks are sufficient to model all of the missions depicted in Figure 1. The next goal is to specify these building blocks as software classes in SHIFT. We have done this for several complex missions, summarized next.

## **Functional architectures**

We have designed architectures for several complex missions. An important point is that despite the diversity of the missions, many of the agents stay the same, i.e. they are "resusable" across missions.

- **Surveillance:** In this mission, illustrated in Figure 3, the "sweeper" is asked to survey a region using helicopters. The sweeper divides the region in several parts, figures out a helicopter trajectory for each region and asks the "dispatcher" to makes sure the helicopters follow those trajectories.



**Figure 4: Surveillance**

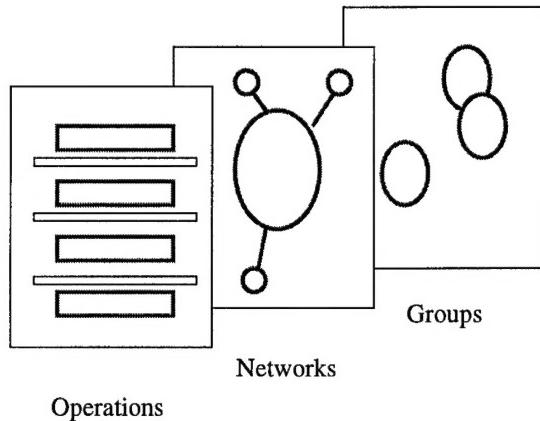


Figure 3. A three-plane realization

The hierarchical architecture has a three-plane realization shown in Figure 3.

The *operations plane* specifies how the agents are coordinated. It gives the information structure and the task structure. The information structure specifies the flow of data between agents and from the environment to the agents via sensors. The task structure specifies who gives tasks to whom and who receives the reports of task execution. The operations plane in effect tells how a mission is to be executed. The other planes play a supporting role.

The *groups plane* specifies the division of agents into groups. Classification of agents into groups is normally carried out with respect to some criteria. Typically these are: capability (eg. vehicles of certain type), location in some area (eg. all helicopters in some sector), availability (eg. all helicopters in some sector that are available for some tasks).

The point of this classification sometimes is that an agent within a certain group may be interchangeable with another agent. This "interchangeability" of agents provides a level of redundancy. Of course, during a mission, agents that are otherwise interchangeable may be assigned different, specialized tasks. Normally, a group of agents is subordinated to a superior agent (under the authority hierarchy) who assigns a task to the group.

The *networks plane* provides the communications infrastructure needed to support operations. Operational needs impose requirements on network support, and limits on network services constrain the information structure that the operations plane can specify. (A different ONR project is devoted to communications infrastructure research.)

The hierarchical, distributed system of the kind depicted in Figure 1 operates in a rapidly changing environment. This environment leads to changes in groups and networks as agents become unavailable because they are tasked or because they are incapacitated. Networks need to be reconfigured as links fail or as the mission progresses. The structure depicted in Figure 2, however, may give a misleading impression of a static organization. During the course of a mission, the assignment of tasks, the group membership,

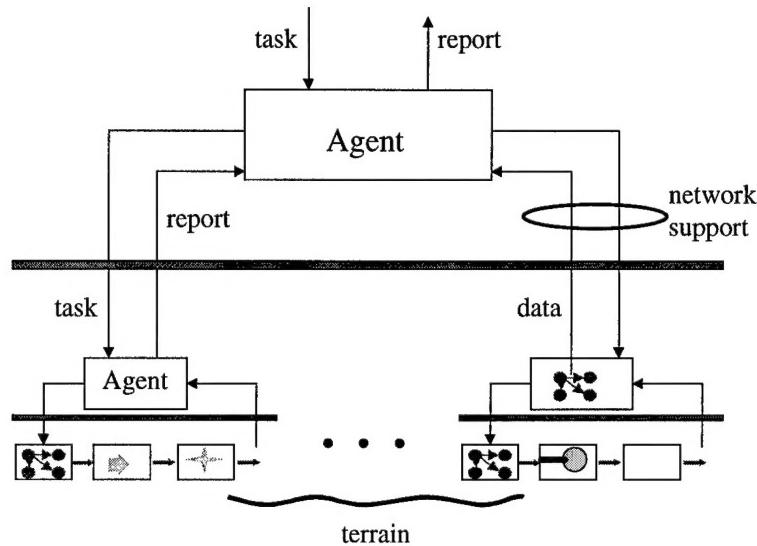


Figure 2. A hierarchical organization of agents

A characteristic feature of hierarchical organizations is that upper layer agents are concerned with more strategic decisions:

- Agents at upper layers have more abstract capabilities than those at lower layers;
- Lower layer capabilities are implemented in fast-acting parallel feedback loops;
- Upper layer capabilities are deliberative, use theoretical concepts to categorize lower layer behavior, and are implemented with different kinds of state space than in feedback control.

This feature arises from two different hierarchical relations:

- Authority hierarchy --- who can task whom; and
- Capability hierarchy --- capabilities of agents at a higher layer are built from lower layer capabilities.

Examples of these hierarchical relations are given for specific scenarios later.

### **The three-plane architecture realization**

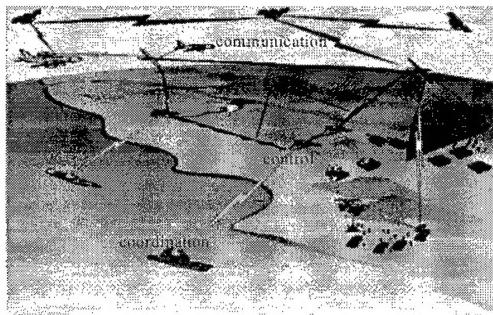


Figure 1. The setting

## **Hierarchical organization**

Since our interest is in decision-making agents, an organization of agents is defined in terms of (1) the *control* that each agent can exercise, (2) the ways in which different agents are *coordinated*, and (3) the *communication* networks that supports the data flow to each agent. Thus the next wave of research must be concerned with

- Architectures for control, coordination, and communication;
- Tools that assist the design, simulation, and implementation of these architectures;
- Demonstrating the proof of concept of organizational intelligence.

A distributed system of agents (or system of systems) is organized in a hierarchy in order to

- limit cognition, control, and communication complexity;
- obtain modular and incremental designs;
- increase adaptation to changes in mission and environment; and
- achieve a scalable design.

Figure 2 depicts a hierarchical organization of agents of the kind suggested in Figure 1.